REMARKS

Applicant respectfully requests reconsideration and allowance of the subject application. Claims 13-24 are pending, of which claims 13-24 have been amended.

35 U.S.C. §101 Claim Rejections

Claims 21-24 are rejected under 35 U.S.C. §101 as being directed to non-statutory subject matter (Office Action p.2). Appropriate amendments to claims 21-24 have been provided herein. Accordingly, claims 21-24 are in condition for allowance and Applicant respectfully requests that the §101 rejection be withdrawn.

35 U.S.C. §102 Claim Rejections

Claims 13-24 are rejected under 35 U.S.C. §102(b) as being anticipated by Aho et al., "Compilers: Principles, Techniques, and Tools" (1986) (hereinafter, "Aho"). Applicant respectfully traverses the rejection.

The Office relies on Aho which is a textbook that describes compilers. A compiler reads a program written in a particular source language and translates it into an equivalent program in a target language, such as another programming language or machine code (Aho p.1, §1.1).

The Office states that "Aho teaches techniques to build a compiler" which is a computer program (Office Action p.4) and that this is a general basis to reject the claims. Although Aho describes aspects of compiler design in Chapter 11 of the textbook, the Office only relies on an introduction to compiling in Chapter 1.

Specifically, the Office refers to three analysis phases of compiling which include lexical, syntax, and semantic analysis (Aho p.12, §1.3). The Office recognizes that a compiler can generate a programming construct into a syntax tree during syntax analysis (Office Action p.4; Aho p.12, §1.3).

The analysis phases of compiling, as well as generating a syntax tree during compiling, are described in the Background section of the present application (p.1, line 19 – p.2, line 18). In contrast to a compiler that translates a program written in a *specific* programming language into another programming language or machine code, Applicant describes that a program can be developed and represented by a "high-level program tree that is a *syntax-independent* representation" of a programmer's intent (Description, p.4, lines 33-37). A programmer can directly manipulate the program tree, "which is in contrast to conventional programming systems in which a programmer manipulates a textual representation of the program that is later converted into a syntax tree during compilation" – as described in Aho. (Description, p.5, lines 1-5).

Claim 13 for example, describes "a first node of a data structure, the first node representing a syntax-independent programming intent". Aho does not show or disclose a node of a data structure "representing a syntax-independent programming intent", as recited in claim 13. Aho only describes that a compiler, which is language specific, generates a syntax tree from a token stream that represents a programming statement during the compiling process (Aho p.12, §1.3). Accordingly, claim 13, as well as dependent claims 14-16, are allowable

over Aho and Applicant respectfully requests that the §102 rejection be withdrawn.

Claim 17 recites a method of handling data comprising "reading a first node of a hierarchical tree, the first node representing a syntax-independent programming intent". Aho does not show or disclose a node of a hierarchical tree "representing a syntax-independent programming intent", as recited in claim 17. As described above in the response to the rejection of claim 13, Aho describes that a compiler generates a syntax tree from a token stream that represents a programming statement during the compiling process (Aho p.12, §1.3). Accordingly, claim 17, as well as dependent claims 18-20, are allowable over Aho and Applicant respectfully requests that the §102 rejection be withdrawn.

Claim 21 recites a data structure comprising "a first node representative of a syntax-independent programming intent". Aho does not show or disclose a node of a data structure that is "representative of a syntax-independent programming intent", as recited in claim 21. As described above in the response to the rejection of claim 13, Aho describes that a compiler generates a syntax tree from a token stream that represents a programming statement during the compiling process (Aho p.12, §1.3). Accordingly, claim 21, as well as dependent claims 22-24, are allowable over Aho and Applicant respectfully requests that the §102 rejection be withdrawn.

Conclusion

Pending claims 13-24 are in condition for allowance. Applicant respectfully requests reconsideration and issuance of the subject application. If any issues remain that preclude issuance of this application, the Examiner is urged to contact the undersigned attorney before issuing a subsequent Action.

Respectfully Submitted,

Dated: May 1, 2003 By: _____

David A. Morasch
Reg. No. 42,905
(509) 324-9256 x 210